AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of Claims:

1-33. (Canceled)

34. (Currently Amended) A method for generating an object schema used in mapping between a relational database and objects from an object oriented programming language comprising: ~~employing a processor coupled to a memory to execute the generation of the object schema, comprising:~~

receiving program code that describes one or more classes which define objects, wherein the objects are components from an object oriented programming language comprising data structures and functions operable on data;

describing members of each class, wherein the members of each class comprise compound members, wherein the compound members comprise a second member and at least one of a plurality of attributes describing the members of each class, and wherein the compound members[[that]] allow mapping of complex members as inline members of a given class, which allows inline mapping of arrays, structs and entity key members;

specifying relationships between the one or more classes;

receiving input from a developer through an interface component;

generating an object schema using the input received from the interface component to be employed to facilitate mapping ~~object components from an object oriented program~~the objects described in the received program code to tables in a relational database, wherein data in the relational database describes the objects and the data in the relational database persists, the object schema comprising:

a first data structure comprising a plurality of attributes describing the one or more classes which define the objects, the plurality of attributes describing the one or more classes comprising at least a persistence service class attribute designating a persistence service to use when persisting a particular class associated with the persistence service class attribute;

a second data structure comprising the plurality of attributes describing the members of each class, the plurality of attributes describing the members of each class comprising at least

a hidden attribute that defines if there is a hidden member in a corresponding class and manages the hidden member in a transparent fashion,

a key generator attribute designating a user class that is to act as a custom key generator, and

a key generator parameter attribute designating parameters to be passed to the custom key generator;

a third data structure comprising a plurality of attributes describing the relationships between the one or more classes, the plurality of attributes describing the relationships between the one or more classes comprising at least a relationship name attribute identifying a unique name for a relationship, and a relationship type attribute identifying a type of predefined relationship; and

wherein at least one of the members described in the second data structure contains an alias attribute to query a private member, the alias attribute pointing to a public member that is to be utilized in place of the associated private member in text of a query;

providing a relational schema that provides details regarding the relational database and utilizes metadata associated with the database to generate an implementation neutral or an implementation specific format that represents the database structure; and

providing a mapping schema that provides a mapping between the object schema and [[a]]the relational schema.~~, and the relational schema utilizes metadata associated with the database to generate an implementation neutral or an implementation specific format that represents the database structure;~~

~~identifying a name of a member to be used as an alias to query a private member, the alias points to a public member that is to be utilized in place of the associated private member in text of a query; and~~

~~defining a hidden member in a corresponding class and managing the hidden member in a transparent fashion.~~

35. (Currently Amended) The method of claim 34, wherein the ~~developer provides~~ input component is [[via]] a graphical user interface.

36. (Canceled)

37. (Original) The method of claim 34, wherein the schema is an XML schema.

38. (Canceled)

39. (Currently Amended) A computer readable storage medium having stored thereon computer executable instructions, which when executed by a processor, ~~for carrying~~ carry out the method of claim 34.

40. (New) A system that facilitates generating an object schema used in mapping between a relational database and objects from an application written in an object-oriented programming language, comprising:

a processor; and

a memory storing computer executable instructions, which when executed by the processor carry out the method of claim 34.

41. (New) The method of claim 34, wherein the plurality of attributes describing the one or more classes further comprises one or more of name, base class, persistence service assembly, and paths.

42. (New) The method of claim 34, wherein the plurality of attributes describing the members of each class further comprises one or more of name, key, key type, type, array length, and search inheritance hierarchy.

43. (New) The method of claim 34, wherein the plurality of attributes describing the relationships between the one or more classes further comprises one or more of parent class, child class, parent member, child member, composition, parent cardinality, child cardinality, and default span.

44. (New) The method of claim 34, wherein the relationship between the one or more classes is one of one-to-one, one-to-many, and many-to-many.

45. (New) The method of claim 34, wherein the relationship between the one or more classes is a parent-child relationship.

46. (New) The method of claim 34, wherein generating the object schema further uses adaptive artificial intelligence technologies including one of a Bayesian network or neural networks to infer proper schema structures and relationships.

47. (New) The method of claim 34, wherein the program code is received in real time as it is being read or transferred en masse upon complete reading of the program code.

48. (New) The method of claim 35, wherein the graphical user interface includes at least one of buttons, text boxes, drop-down menus, tabs, hyperlinks, check boxes, and scroll bars.

49. (New) The method of claim 48, wherein the graphical interface comprises tabs and the tabs are selected from a list of assembly, classes, inheritance, members, hidden members, keys, aliases, and relationships.

50. (New) The method of claim 48, wherein the graphical interface comprises tabs and the tabs are selected from a list of object schema document (OSD) builder and extensible markup language (XML) output.

51. (New) The method of claim 34, wherein the interface component is a wizard that specifies a series of steps and each step must be completed before the developer is allowed to advance to a next step in the series of steps.